EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	20136080	@ad<"20001228"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:25
L2	0	"711"/$.ccls	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:26
L4	30083	"711"/$.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:26
L5	73161	"365"/$.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:26
L6	4237	request$4 adj node	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L7	1951	MESI	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L8	7560	Snoop$4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L9	10631	buffer near2 full	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L10	141	L6 and L7	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27

EAST Search History

L11	596	L8 and L9	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L12	1	L10 and L11	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L13	35	(Akhilesh adj Kumar).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L14	1	L10 and L11	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L15	0	L14 and L13	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L16	646	(before or prior) adj2 coheren$4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L17	0	((read$3 adj2 (before or prior)) adj2 coheren$4) near4 advant$5	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:27
L18	11	((read$3 adj2 (before or prior)) adj2 coheren$4)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:28
L19	208	(speculat$6 same read same coheren$5)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:28
L20	69183	request$5 adj2 (node$1 or agent$2 or process$6)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:29

EAST Search History

L21	97	19 and 20	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:29
L22	6098	cache adj coheren$5	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:30
L23	85	21 and 22	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:30
L24	10295	buffer near (full or empty)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:30
L25	1	23 and 24	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/07/31 20:30

Terms used
speculative read request coherency MESI buffer full cache coherency

Found **7** of **182,223**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 7 of 7

Relevance scale

1 Hardware prediction for data coherency of scientific codes on DSM

J. T. Acquaviva, W. Jalby

November 2000 **Proceedings of the 2000 ACM/IEEE conference on Supercomputing (CDROM)**

Publisher: IEEE Computer Society

Full text available: pdf(142.06 KB) Publisher Site

Additional Information: full citation, abstract, references, index terms

This paper proposes a hardware mechanism for reducing coherency overhead occurring in scientific computations within DSM systems. A first phase aims at detecting, in the address space regular patterns (called streams) of coherency events (such as requests for exclusive, shared or invalidation). Once a stream is detected at a loop level, regularity of data access can be exploited at the loop level (spatial locality) but also between loops (temporal locality). We present a hardwa ...

2 Micro-architecture techniques in the intel® E8870 scalable memory controller

Fayé Briggs, Suresh Chittor, Kai Cheng

June 2004 **Proceedings of the 3rd workshop on Memory performance issues: in conjunction with the 31st international symposium on computer architecture WMPI '04**

Publisher: ACM Press

Full text available: pdf(435.63 KB) Additional Information: full citation, abstract, references, index terms

This paper describes several selected micro-architectural tradeoffs and optimizations for the scalable memory controller of the Intel E8870 chipset architecture. The Intel E8870 chipset architecture supports scalable coherent multiprocessor systems using 2 to 16 processors, and a point-to-point Scalability Port (SP) Protocol. The scalable memory controller micro-architecture applies a number of micro-architecture techniques to reduce the local & remote idle and loaded latencies. The performance ...

Keywords: distributed coherency, memory latency, scalability, transaction flows

3 Parallel architectures: Inferential queueing and speculative push for reducing critical communication latencies

Ravi Rajwar, Alain Kägi, James R. Goodman

June 2003 **Proceedings of the 17th annual international conference on**

Supercomputing
Publisher: ACM Press
Full text available: pdf(568.93 KB) Additional Information: full citation, abstract, references, index terms

Communication latencies within critical sections constitute a major bottleneck in some classes of emerging parallel workloads. In this paper, we argue for the use of Inferentially Queued Locks (IQLs) [31], not just for efficient synchronization but also for reducing communication latencies, and we propose a novel mechanism, Speculative Push (SP), aimed at reducing these communication latencies. With IQLs, the processor infers the existence, and limits, of a critical section from the use of synch ...

Keywords: data forwarding, inferential queueing, synchronization

4 Multithreading and value prediction: Speculative lock elision: enabling highly concurrent multithreaded execution
Ravi Rajwar, James R. Goodman
December 2001 **Proceedings of the 34th annual ACM/IEEE international symposium on Microarchitecture**
Publisher: IEEE Computer Society
Full text available: pdf(1.37 MB) Additional Information: full citation, abstract, references, citings
Publisher Site

Serialization of threads due to critical sections is a fundamental bottleneck to achieving high performance in multithreaded programs. Dynamically, such serialization may be unnecessary because these critical sections could have safely executed concurrently without locks. Current processors cannot fully exploit such parallelism because they do not have mechanisms to dynamically detect such false inter-thread dependences.We propose *Speculative Lock Elision (SLE),* a novel micro-architectura ...

5 Piranha: a scalable architecture based on single-chip multiprocessing
Luiz André Barroso, Kourosh Gharachorloo, Robert McNamara, Andreas Nowatzyk, Shaz Qadeer, Barton Sano, Scott Smith, Robert Stets, Ben Verghese
May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture ISCA '00**, Volume 28 Issue 2
Publisher: ACM Press
Full text available: pdf(191.10 KB) Additional Information: full citation, abstract, references, citings, index terms

The microprocessor industry is currently struggling with higher development costs and longer design times that arise from exceedingly complex processors that are pushing the limits of instruction-level parallelism. Meanwhile, such designs are especially ill suited for important commercial applications, such as on-line transaction processing (OLTP), which suffer from large memory stall times and exhibit little instruction-level parallelism. Given that commercial applications constitute by fa ...

6 Performance of database workloads on shared-memory systems with out-of-order processors
Parthasarathy Ranganathan, Kourosh Gharachorloo, Sarita V. Adve, Luiz André Barroso
October 1998 **ACM SIGPLAN Notices , ACM SIGOPS Operating Systems Review , Proceedings of the eighth international conference on Architectural support for programming languages and operating systems ASPLOS-VIII**, Volume 33 , 32 Issue 11 , 5
Publisher: ACM Press
Full text available: pdf(1.62 MB) Additional Information: full citation, abstract, references, citings, index terms

Database applications such as online transaction processing (OLTP) and decision support systems (DSS) constitute the largest and fastest-growing segment of the market for multiprocessor servers. However, most current system designs have been optimized to perform well on scientific and engineering workloads. Given the radically different behavior of database workloads (especially OLTP), it is important to re-evaluate key system design decisions in the context of this important class of applicatio ...

7 Interconnect-Aware Coherence Protocols for Chip Multiprocessors

Liqun Cheng, Naveen Muralimanohar, Karthik Ramani, Rajeev Balasubramonian, John B. Carter

June 2006 **Proceedings of the 33rd International Symposium on Computer Architecture ISCA '06**

Publisher: IEEE Computer Society

Full text available: pdf(367.59 KB) Additional Information: full citation, abstract

Improvements in semiconductor technology have made it possible to include multiple processor cores on a single die. Chip Multi-Processors (CMP) are an attractive choice for future billion transistor architectures due to their low design complexity, high clock frequency, and high throughput. In a typical CMP architecture, the L2 cache is shared by multiple cores and data coherence is maintained among private L1s. Coherence operations entail frequent communication over global on-chip wires. In fut ...

Results 1 - 7 of 7

IEEE Xplore RELEASE 2.1

Mon, 31 Jul 2006, 9:19:17 PM EST

Edit an existing query or compose a new query in the Search Query Display.

Select a search number (#) to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

Search Query Display

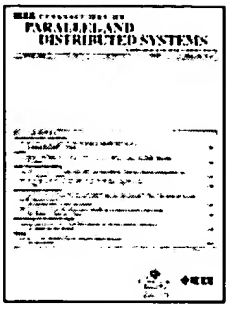
[redacted]

Recent Search Queries

#1	((cache and coheren*)<in>metadata)
#2	(speculat* and memory read and request<IN>metadata)
#3	(buffer or queue or register) and (empty or full)
#4	(read and ahead<IN>metadata)
#5	(snoop* and (read* or writ*)<IN>metadata)
#6	(MESI and protocol<IN>metadata)
#7	(((cache and coheren*)<in>metadata)) <AND> ((speculat* and memory read and request<IN>metadata))
#8	((buffer or queue or register) and (empty or full)) <AND> ((read and ahead<IN>metadata))
#9	(((buffer or queue or register) and (empty or full)) <AND> ((read and ahead<IN>metadata))) <AND> ((snoop* and (read* or writ*)<IN>metadata))

[redacted]

Indexed by Inspec



IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS

Year: 2006 | Issue: Volume 17 , Issue 8

Search This Publication

All Fields

IEEE COMPUTER SOCIETY

RSS Feed for Latest Issue: RSS » Learn more

Frequency: 12

ISSN: 1045-9219

Subject Category: Distributed Computing/Real-Time Systems

Published by: IEEE Computer Society

Visit the Website: IEEE Transactions on Parallel and Distributed Systems

OPAC Link: http://ieeexplore.ieee.org/servlet/opac?punumber=71 » Learn more

Publication Information
- Editorial Board

Announcements
- Content Announcements

Author Resources
- IEEE Information
- Addtional Information

Aims and Scope

IEEE Transactions on Parallel and Distributed Systems (*TPDS*) is published monthly. The goal of *TPDS* is to publish a range on previously published papers, and survey articles that deal with the research areas of current importance to our readers. Cu particular interest include, but are not limited to the following: a) architectures: design, analysis, and implementation of multipl (including multi-processors, multicomputers, and networks); impact of VLSI on system design; interprocessor communication languages and compilers; scheduling and task partitioning; databases, operating systems, and programming environments fo systems; c) algorithms and applications: models of computation; analysis and design of parallel/distributed algorithms; applica in better multiple-processor systems; d) other issues: performance measurements, evaluation, modeling and simulation of mu systems; real-time, reliability and fault-tolerance issues; conversion of software from sequential-to-parallel forms.

Contacts

Assistant
TPDS Transactions Assistant
IEEE Computer Society
PO Box 3014
Los Alamitos, CA 90720-1314, USA
Email: tpds@computer.org

Editor-in-Chief
Laxmi Bhuyan
University of California, Riverside
Computer Science and Engineering
319 Surge Bldg
Riverside, CA 92521, USA
Phone: +1 951 827 2347
Fax: +1 951 827 4643
Email: bhuyan@cs.ucr.edu